



⑮ BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENTAMT

⑫ **Gebrauchsmuster**  
⑩ **DE 296 22 133 U 1**

⑲	Aktenzeichen:	296 22 133.3
⑳	Anmeldetag:	19. 12. 96
㉑	Eintragungstag:	17. 7. 97
㉒	Bekanntmachung im Patentblatt:	28. 8. 97

⑤① Int. Cl.<sup>8</sup>:  
**G 05 B 19/05**  
G 05 B 19/414  
G 06 F 13/12  
G 06 F 9/445  
// H04L 12/40

DE 296 22 133 U 1

⑦③ Inhaber:  
Siemens AG, 80333 München, DE

⑥④ Automatisierungsgerät

DE 296 22 133 U 1

19.12.96

1

## Beschreibung

## Automatisierungsgerät

- 5 Die Erfindung betrifft ein Automatisierungsgerät, welchem Software-Funktionsbausteine eines Steuerprogramms zuführbar sind, welches das Automatisierungsgerät während eines Steuerbetriebs zyklisch und/oder interruptgesteuert bearbeitet, wobei die Software-Funktionsbausteine ladbar und zur Laufzeit  
10 des Steuerprogramms in dieses einbindbar ausgebildet sind.

Ein Automatisierungsgerät mit den Merkmalen des Oberbegriffs des Anspruchs 1 ist aus dem Siemens-Katalog ST 70, Ausgabe 1995, bekannt. Ein Anwender erstellt mit einem Programmiergerät ein Programm zur Steuerung eines technischen Prozesses,  
15 das Software-Funktionsbausteine, z. B. in Form von Organisationsbausteinen, Programmbausteinen und Instanzdatenbausteinen, umfaßt. Das Automatisierungsgerät ist mit dem Programmiergerät über ein Bussystem verbunden, über welches das  
20 Programmiergerät das Steuerprogramm in das Automatisierungsgerät überträgt.

Häufig ist es erforderlich, daß Software-Funktionsbausteine eines Steuerprogramms von einem Automatisierungsgerät eines  
25 Fertigungsstandortes oder aus einem Software-Pool dieses Fertigungsstandortes in ein Automatisierungsgerät eines anderen Fertigungsstandortes zu übertragen sind. Insbesondere wenn die Fertigungsstandorte sehr weit voneinander entfernt sind, z. B. wegen einer Globalisierung von Fertigungsaktivitäten, werden diese Software-Funktionsbausteine über das  
30

19.12.96

2

globale Netzwerk „INTERNET“ übertragen. Dazu sind Server mit geeigneten Kommunikationsschnittstellen notwendig, die einerseits das INTERNET-Kommunikationsprotokoll und andererseits das Kommunikationsprotokoll der Automatisierungsgeräte ermöglichen. Aufgrund dieser unterschiedlichen Protokolle und der Architektur der Automatisierungsgeräte ist eine Einbindung der Software-Funktionsbausteine zur Laufzeit des Steuerprogramms nicht möglich, insbesondere dann nicht, wenn Automatisierungsgeräte unterschiedlicher Hersteller mit diesen Software-Funktionsbausteinen zu versorgen sind.

In der deutschen Gebrauchsmusteranmeldung 296 00 609.2 wird ein Automatisierungsgerät für einen Einsatz in einem global verteilten Automatisierungsverbund vorgeschlagen. Dazu sind die Software-Funktionsbausteine des Automatisierungsgerätes objektorientiert ausgebildet und über das INTERNET und eine INTERNET-Kommunikationsschnittstelle des Automatisierungsgerätes in dieses ladbar und ferner ist dazu das Automatisierungsgerät versehen mit einem Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System), das zur Einbindung der Software-Funktionsbaustein-Objekte und zur Bearbeitung des Steuerprogramms vorgesehen ist.

Der vorliegenden Erfindung liegt die Aufgabe zugrunde, ein Automatisierungsgerät der eingangs genannten Art zu schaffen, welches für einen Einsatz in einem global verteilten Automatisierungsverbund geeignet ist. Darüber hinaus ist ein Programmiergerät sowie ein intelligentes Feldgerät anzugeben, welche für einen Einsatz in einem global verteilten Automatisierungsverbund geeignet sind.

19.12.98

3

Diese Aufgabe wird im Hinblick auf das Automatisierungsgerät durch ein Automatisierungsgerät der eingangs genannten Art mit den im kennzeichnenden Teil des Anspruchs 1 angegebenen, im Hinblick auf ein Programmiergerät bzw. intelligentes Feld-  
5 gerät durch die in den Merkmalen des Anspruchs 5 bzw. des Anspruchs 9 angegebenen Maßnahmen gelöst.

Vorteilhaft ist, daß das Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System) für das Automatisierungs-  
10 gerät, das Programmiergerät sowie für das intelligente Feldgerät einheitlich ist, wodurch ein durchgängiges Automatisierungssystem geschaffen wird. Darüber hinaus sind Steuerprogramme einfach zu modellieren und zu implementieren.

15 Vorteilhafte Ausgestaltungen der Erfindung ergeben sich aus den Unteransprüchen.

Anhand der Zeichnung, in der ein Ausführungsbeispiel der Erfindung dargestellt ist, werden nachfolgend die Erfindung,  
20 deren Ausgestaltungen und Vorteile näher erläutert.

Es zeigen

Figur 1 ein universelles, verteiltes Automatisierungs- und Management-Engineering- und Informations-System in schematischer Darstellung und  
25

Figuren 2 und 3 Software-Funktionsbaustein-Ablaufsysteme (PLC-Object-Engine-Systeme).

19.12.96

4

Zwei Fertigungsstandorte 1 und 2 eines global verteilten Automatisierungsverbundes sind über ein an sich bekanntes globales Netzwerk „INTERNET“ 3 miteinander verbunden, wobei geeignete Einrichtungen 4, 5 vorgesehen sind, die verhindern, daß Unberechtigte Daten in die datenverarbeitenden Komponenten der Fertigungsstandorte 1, 2 übertragen. Die Fertigungsstandorte 1, 2 umfassen mehrere jeweils mit einer INTERNET-Kommunikationsschnittstelle versehene Geräte in Form von Automatisierungsgeräten 6, Programmiergeräten 7, Bedien- und Beobachtungsgeräten 8 und Workstations 9. Diese INTERNET-Kommunikationsschnittstellen ermöglichen eine TCP/IP-Protokoll-Kommunikation der Geräte untereinander. Es ist eine wesentliche Anforderung an ein Automatisierungsgerät, das während eines Steuerbetriebs ein aus mehreren Software-Funktionsbausteinen gebildetes Steuerprogramm zyklisch und/oder interruptgesteuert bearbeitet, daß diese Software-Funktionsbausteine ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet sind. Damit diese Anforderung erfüllt ist und die Software-Funktionsbausteine über das INTERNET und die INTERNET-Kommunikationsschnittstelle direkt in ein Automatisierungsgerät ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar sind, sind die Software-Funktionsbausteine objektorientiert ausgebildet. Die Software-Funktionsbausteine sind über das INTERNET dynamisch ladbar und erweiterbar, und das Automatisierungsgerät ist mit einer Software-Funktionsbaustein-Ablaufsteuerung (PLC-Object-Engine-System) versehen, die diese Software-Funktionsbausteine in das Steuerprogramm einbindet und während des Steuerbetriebs bearbeitet.

19.12.95

5

Eine Programmiersprache, die einen objektorientierten Code aus einer Quellsprache erzeugt und für einen Einsatz im INTERNET vorgesehen ist, ist aus dem Buch „Java!“, Tim Ritchey, published 1995 by New Riders Publishing, bekannt. Dort ist eine Quellsprache „JAVA C“ beschrieben, aus der ein objektorientierter Java-Bytecode erzeugbar ist. Weitere vorteilhafte Eigenschaften dieses Sprachmittels sind insbesondere die Portabilität des Codes sowie die Mechanismen zur Fehlerbehandlung. Durch die Portabilität des Codes wird sichergestellt, daß ein Automatisierungsgerät mit einer Ablaufsteuerung in Form eines Java-Bytecode-Interpreters unabhängig von einer Prozessor-Hardware-Architektur des Automatisierungsgerätes (herstellerunabhängig) die dem Automatisierungsgerät über das INTERNET zugeführten Java-Funktionsbausteine bearbeiten kann. Aus Gründen der Performance ist es allerdings vorteilhaft, das Automatisierungsgerät mit einem Java-Prozessor zu versehen, der den Java-Code direkt verarbeitet.

Die Programmierung der objektorientierten Software-Funktionsbausteine erfolgt durch die jeweiligen Programmiergeräte (Figur 1) der Fertigungsstandorte 1, 2 oder durch ein ebenfalls an das INTERNET angeschlossenes Programmiergerät 14. Neben den Bedien- und Beobachtungsgeräten 8 und den Workstations 9 sind diese Programmiergeräte 8, 14 Bestandteile des Management-Engineering-Systems. Die Programmiergeräte führen diese Softwarebausteine den entsprechenden Automatisierungsgeräten über die jeweilige INTERNET-Kommunikationsschnittstelle und das INTERNET zu. Für den Fall, daß z. B. Bausteine geändert werden müssen, überträgt zunächst das

19.12.96

6

Automatisierungsgerät 6 oder ein Server 13 einem der Programmiergeräte 7 den entsprechenden Software-Funktionsbaustein über das INTERNET. Schließlich ergänzt bzw. modifiziert das Programmiergerät 7 diesen Baustein und kann ihn wieder in  
5 eines der Automatisierungsgeräte übertragen. Das Programmiergerät ist ferner mit einem Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE, Wd, IO) versehen, das zur Simulation des Steuerprogramms vorgesehen ist.

10 Um in Automatisierungssystemen hohen Ausbaugrades die Anzahl der in ein Automatisierungsgerät eingebauten Ein- und Ausgabekomponenten zu verringern, werden dezentrale Subsysteme, z. B. in Form von intelligenten Feldgeräten, eingesetzt. Das verteilte Automatisierungs- und Management-Engineering-System  
15 weist ein hier nicht dargestelltes intelligentes Feldgerät auf, welchem mindestens ein Software-Funktionsbaustein eines Steuerprogramms zuführbar ist, welches das Feldgerät während eines Steuerbetriebs zyklisch und/oder interruptgesteuert bearbeitet, wobei der Software-Funktionsbaustein ladbar und  
20 zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet ist. Die Software-Funktionsbausteine sind objektorientiert ausgebildet und über das INTERNET und eine INTERNET-Kommunikationsschnittstelle des Feldgerätes in dieses ladbar, wobei das Feldgerät ein Software-Funktions-  
25 baustein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE, Wd, IO) zur Einbindung des Software-Funktionsbausteins SF01 ... SF04 und Bearbeitung des Steuerprogramms aufweist.

Im folgenden wird auf Figur 2 und 3 verwiesen, in denen ein  
30 Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-

19.12.95

7

- System) eines Automatisierungsgerätes und/oder eines intelligenten Feldgerätes und/oder eines Programmiergerätes (zur Simulation eines Steuerprogramms) dargestellt ist. Es ist zunächst angenommen, daß ein Steuerprogramm zyklisch zu be-
- 5 arbeiten ist (Figur 2), was bedeutet, daß unabhängig von den Signalzuständen von Prozeßein- und -ausgängen eines zu steuernden technischen Prozesses z. B. die CPU eines Automatisierungsgerätes zyklisch
- a. die Signalzustände der Prozeßeingänge abfragt und in einem
  - 10 Prozeßabbild der Eingänge hinterlegt,
  - b. entsprechend den Vorgaben des zu bearbeitenden Steuerprogramms dieses schrittweise abarbeitet und
  - c. die errechneten Signalzustände in einem Prozeßabbild der
  - 15 Ausgänge hinterlegt, wobei diese Signalzustände von dort zu den Prozeßausgängen gelangen.

Wesentliche Bestandteile des Software-Funktionsbaustein-Ablaufsystems sind objektorientiert programmierte Einheiten in Form eines Bootstraps Bos, eines Ein-/Ausgabe-Moduls IO,

20 eines Exe-Engine-Objekts ExE und eines Watchdogs Wd. Der Watchdog Wd braucht selbstverständlich nicht als Softwaremodul ausgebildet sein, sondern kann hardwaremäßig verwirklicht werden. In einem praktischen Ausführungsbeispiel der Erfindung sind die Einheiten Exe-Engine-Objekt ExE und

25 Watchdog Wd sogenannte „threads“. Die Funktions- und Wirkungsweise eines „threads“ ist aus der Druckschrift „Supporting Microsoft Windows 95, Student Workbook“, 07/95, der Fa. Microsoft bekannt und braucht daher nicht näher erläutert zu werden. In der Bootstrap-Einheit Bos sind eine

30 Klasse von Software-Funktionsbausteinen und eine Klasse von



19.12.95

8

Ein-/Ausgabe-Modulen hinterlegt. Diese Klassen werden z. B. von einem Anwender entsprechend den Vorgaben einer zu lösenden Steuerungsaufgabe auf einem Programmiergerät erstellt und z. B. in ein Automatisierungsgerät oder in ein Feldgerät übertragen. Die Bootstrap-Einheit Bos erzeugt vor Beginn des Steuerbetriebes aus der Klasse Software-Funktionsbausteine Software-Funktionsbaustein-Objekte und aus der Klasse Ein-/Ausgabe-Module Ein-/Ausgabe-Modul-Objekte. Im vorliegenden Beispiel sind lediglich vier Software-Funktionsbaustein-Objekte SFO1 ... SFO4 und ein Ein-/Ausgabe-Modul-Objekt IO dargestellt, in welchem ein Prozeßabbild von Ein- und Ausgängen hinterlegt ist und welchem Signalzustände von Prozeßeingängen des technischen Prozesses zuführbar sind und durch welches Signalzustände Prozeßausgängen dieses technischen Prozesses zuführbar sind. Ferner führt die Bootstrap-Einheit Bos zu Beginn des Steuerbetriebes dem Exe-Engine-Objekt ExE eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte SFO1 ... SFO4 zu. Zu Beginn des Steuerbetriebes überträgt die Bootstrap-Einheit Bos eine Nachricht Nas (Methodenaufruf), wodurch das Exe-Engine-Objekt ExE gestartet wird. In einem ersten Bearbeitungsschritt führt das Exe-Engine-Objekt ExE dem Watchdog Wd eine Nachricht Naw zu, was bewirkt, daß der Watchdog Wd die Zykluszeit des Exe-Engine-Objekts ExE überwacht. Für den Fall, daß das Exe-Engine-Objekt ExE die vorgesehene Zykluszeit überschreitet, setzt der Watchdog Wd das Exe-Engine-Objekt ExE zurück, indem der Watchdog Wd dem Exe-Engine-Objekt ExE eine Nachricht Nar überträgt. Ferner setzt der Watchdog Wd im Falle der Zykluszeitüberschreitung die Ausgänge des Prozeßabbildes und die Prozeßausgänge zurück, wobei der Watchdog Wd dazu dem

19.12.96

9

Ein-/Ausgabe-Modul-Objekt IO eine Nachricht Nia zuführt.  
Nachdem das Exe-Engine-Objekt ExE den Watchdog Wd gestartet  
hat, nimmt das Exe-Engine-Objekt ExE den Steuerbetrieb auf  
und führt zunächst einen Bearbeitungszyklus aus, welcher die

5 Schritte umfaßt:

- A) Aktualisieren der Eingänge des Prozeßabbildes, indem das  
Exe-Engine-Objekt ExE die Eingangssignalzustände des  
Prozeßabbildes aus dem Ein-/Ausgabe-Modul-Objekt IO aus-  
liest, wobei die Signalzustände durch einen Methodenaufruf  
10 Nae zwischen den Objekten ausgetauscht werden,
- B) Bearbeiten jeweils eines Bearbeitungsschrittes der Soft-  
ware-Funktionsbaustein-Objekte SFO1 ... SFO4, wobei das  
Exe-Engine-Objekt ExE entsprechende Methodenaufrufe NSF1  
... NSF4 den Software-Funktionsbaustein-Objekten SFO1 ...  
15 SFO4 zuführt, die Software-Funktionsbaustein-Objekte SFO1  
... SFO4 über Methodenaufrufe Nso auf das Prozeßabbild des  
Ein-/Ausgabe-Modul-Objekts IO zugreifen, und schließlich
- C) Aktualisieren der Ausgänge des Prozeßabbildes, indem das  
Exe-Engine-Objekt ExE die Ausgänge des Prozeßabbildes im  
20 Ein-/Ausgabe-Modul-Objekt IO einschreibt, wobei wiederum  
die Signalzustände durch einen Methodenaufruf Naa zwischen  
den Objekten ausgetauscht werden.

Das Ein-/Ausgabe-Modul-Objekt IO versorgt über geeignete  
25 Schnittstellen die Prozeßausgänge des zu steuernden techni-  
schen Prozesses und - wie beschrieben - die Software-Funk-  
tionsbaustein-Objekte SFO1 ... SFO4 mit den Ein- und Aus-  
gangssignalzuständen des Prozeßabbildes.

19.12.96

10

Es wird nun der Fall angenommen, daß ein Steuerprogramm interruptgesteuert zu bearbeiten ist (Figur 3), was bedeutet, daß im Falle von Signalzustandsänderungen an einem der Prozeßeingänge eines zu steuernden technischen Prozesses unverzüglich geeignete Steuermaßnahmen zu ergreifen sind. Die in den Figuren 2 und 3 gleichen Teile sind mit gleichen Bezugszeichen versehen. Im folgenden werden nur die Unterschiede zu der zyklischen Bearbeitung des Steuerprogramms beschrieben (Figur 2). Im Falle der interruptgesteuerten Bearbeitung des Steuerprogramms überträgt die Bootstrap-Einheit Bos zu Beginn des Steuerbetriebs dem Exe-Engine-Objekt ExE nicht wie im Falle einer zyklischen Bearbeitung die Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte, sondern eine Liste von „aufzurufenden“ Software-Funktionsbaustein-Objekten pro Prozeßeingang. Das bedeutet, jedem Prozeßeingang ist ein Software-Funktionsbaustein-Objekt zugeordnet und bei einer Signalzustandsänderung an einem Prozeßeingang ist das diesem Eingang zugeordnete Software-Funktionsbaustein-Objekt zu starten. Im Unterschied zur zyklischen Bearbeitung im Hinblick auf den Bearbeitungsschritt B) führt das Exe-Engine-Objekt ExE während des Bearbeitungszyklus einen Schritt B\*) aus, welcher umfaßt:

B\*) Feststellen von Signalzustandsänderungen an den Eingängen des Prozeßabbildes und Bearbeiten der diesen Eingängen zugeordneten Software-Funktionsbaustein-Objekte.

Durch die Erfindung wird ein durchgängiges Automatisierungssystem geschaffen, das einen weltweiten Betrieb ermöglicht. Die aus dem Stand der Technik bekannte weitgehende Entkopp-

19.12.95

11

lung von Automatisierungsgeräten und Standard-Computern wird  
vermieden.

This Page Blank (uspic)

19.12.95

12

## Schutzansprüche

1. Automatisierungsgerät, welchem Software-Funktionsbausteine eines Steuerprogramms zuführbar sind, welches das Automati-
- 5 sierungsgerät während eines Steuerbetriebs zyklisch und/oder interruptgesteuert bearbeitet, wobei die Software-Funktionsbausteine ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet sind, dadurch gekennzeichnet,
- 10 - daß die Software-Funktionsbausteine (SF01 ... SF04) objektorientiert ausgebildet und über das INTERNET und eine INTERNET-Kommunikationsschnittstelle des Automatisierungsgerätes in dieses ladbar sind und
- daß das Automatisierungsgerät ein Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE, Wd,
- 15 IO) zur Einbindung der Software-Funktionsbaustein-Objekte (SF01 ... SF04) und zur Bearbeitung des Steuerprogramms aufweist, wobei
- das Software-Funktionsbaustein-Ablaufsystem ein Exe-
- 20 Engine-Objekt (ExE), einen Watchdog (Wd), einen Bootstrap (Bos) und ein Ein-/Ausgabe-Modul-Objekt (IO) umfaßt, in welchem ein Prozeßabbild von Ein- und Ausgängen hinterlegbar ist und welchem Signalzustände von Prozeßeingängen zuführbar sind und durch welches
- 25 Signalzustände Prozeßausgängen zuführbar sind,
- der Bootstrap (Bos) vor Beginn des Steuerbetriebes die Software-Funktionsbaustein-Objekte (SF01 ... SF04) und das Ein-/Ausgabe-Modul-Objekt (IO) erzeugt sowie dem Exe-Engine-Objekt (ExE) zuführt:

19.12.95

13

- für den Fall einer zyklischen Bearbeitung des Steuerprogramms eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte (SF01 ... SF04),
- 5       - für den Fall einer interruptgesteuerten Bearbeitung des Steuerprogramms eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte (SF01 ... SF04) für jeden Prozeßeingang,
- der Bootstrap (Bos) zu Beginn des Steuerbetriebes das  
10       Exe-Engine-Objekt (ExE) startet, welches zunächst den Watchdog (Wd) startet, welcher beim Überschreiten der Zykluszeit das Exe-Engine-Objekt (ExE) zurücksetzt, und anschließend zyklisch
  - die Eingänge des Prozeßabbildes aktualisiert,
  - 15       - für den Fall einer zyklischen Bearbeitung des Steuerprogramms jeweils einen Bearbeitungsschritt der Software-Funktionsbaustein-Objekte (SF01 ... SF04) bearbeitet,
  - für den Fall einer interruptgesteuerten Bearbeitung  
20       des Steuerprogramms Änderungen von Signalzuständen an den Eingängen feststellt und die diesen Eingängen zugeordneten Software-Funktionsbaustein-Objekte (SF01 ... SF04) bearbeitet,
  - die Ausgänge des Prozeßabbildes aktualisiert.

25

2. Automatisierungsgerät nach Anspruch 1, dadurch gekennzeichnet, daß das Exe-Engine-Objekt (ExE) sowie der Watchdog (Wd) als „threads“ ausgebildet sind.

19.12.96

14

3. Automatisierungsgerät nach Anspruch 1 oder 2, dadurch gekennzeichnet, daß die Kommunikationsschnittstelle eine TCP/IP-Protokoll-Kommunikation ermöglicht.

- 5 4. Automatisierungsgerät nach einem der Ansprüche 1 bis 3, dadurch gekennzeichnet, daß die Software-Funktionsbausteine (SF01 ... SF04) Java-bytecodiert sind und in der Programmiersprache „JAVA C“ oder in einer Programmiersprache nach der Norm IEC 1131 erstellbar sind.

10

5. Programmiergerät zur Erstellung von Software-Funktionsbausteinen eines Steuerprogramms, das einem Automatisierungsgerät zuführbar ist, welches während eines Steuerbetriebs das Steuerprogramm zyklisch und/oder interruptgesteuert bearbeitet, wobei die Software-Funktionsbausteine ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet sind,

- dadurch gekennzeichnet,
- daß das Programmiergerät die Software-Funktionsbausteine (SF01 ... SF04) objektorientiert ausgebildet erstellt,
  - daß das Programmiergerät dem Automatisierungsgerät über das INTERNET und eine INTERNET-Kommunikationsschnittstelle des Programmiergerätes die Software-Funktionsbausteine (SF01 ... SF04) zuführt und/oder
  - daß dem Programmiergerät über das INTERNET und die INTERNET-Kommunikationsschnittstelle die Software-Funktionsbausteine (SF01 ... SF04) zuführbar sind.

6. Programmiergerät nach Anspruch 5, dadurch gekennzeichnet, daß das Programmiergerät zur Simulation des

19.12.95

15

Steuerprogramms ein Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE, Wd, IO) aufweist.

7. Programmiergerät nach Anspruch 5 oder 6, dadurch gekennzeichnet, daß die Kommunikationsschnittstelle eine TCP/IP-Protokoll-Kommunikation ermöglicht.

8. Programmiergerät nach einem der Ansprüche 5 bis 7, dadurch gekennzeichnet, daß die Software-Funktionsbausteine (SF01 ... SF04) in der auf dem Programmiergerät ablauffähigen Programmiersprache „JAVA C“ oder in einer Programmiersprache nach der Norm IEC 1131 erstellbar und durch das Programmiergerät Java-bytecodiert übersetzbar sind.

9. Intelligentes Feldgerät, welchem mindestens ein Software-Funktionsbaustein eines Steuerprogramms zuführbar ist, welches das Feldgerät während eines Steuerbetriebs zyklisch und/oder interruptgesteuert bearbeitet, wobei der Software-Funktionsbaustein ladbar und zur Laufzeit des Steuerprogramms in dieses einbindbar ausgebildet ist, dadurch gekennzeichnet,

- daß der Software-Funktionsbaustein (SF01 ... SF04) objektorientiert ausgebildet und über das INTERNET und eine INTERNET-Kommunikationsschnittstelle des Feldgerätes in dieses ladbar ist und

- daß das Feldgerät ein Software-Funktionsbaustein-Ablaufsystem (PLC-Object-Engine-System; Bos, ExE, Wd, IO) zur Einbindung des Software-Funktionsbausteins (SF01 ... SF04) und Bearbeitung des Steuerprogramms aufweist.



19.12.95

16

10. Intelligentes Feldgerät nach Anspruch 9, dadurch gekennzeichnet,

- daß das Software-Funktionsbaustein-Ablaufsystem ein Exe-Engine-Objekt (ExE), einen Watchdog (Wd), einen Bootstrap (Bos) und ein Ein-/Ausgabe-Modul-Objekt (IO) umfaßt, in  
5       welchem ein Prozeßabbild von Ein- und Ausgängen hinterlegbar ist und welchem Signalzustände von Prozeßeingängen zu-  
führbar sind und durch welche Signalzustände Prozeß-  
ausgängen zuführbar sind,
- 10   - daß der Bootstrap (Bos) vor Beginn des Steuerbetriebes die Software-Funktionsbaustein-Objekte (SF01 ... SF04) und das Ein-/Ausgabe-Modul-Objekt (IO) erzeugt sowie dem Exe-Engine-Objekt (ExE) zuführt:
  - für den Fall einer zyklischen Bearbeitung des Steuer-  
15       programms eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte (SF01 ... SF04),
  - für den Fall einer interruptgesteuerten Bearbeitung des Steuerprogramms eine Liste der zu bearbeitenden Software-Funktionsbaustein-Objekte (SF01 ... SF04) für  
20       jeden Prozeßeingang,
- daß der Bootstrap (Bos) zu Beginn des Steuerbetriebes das Exe-Engine-Objekt (ExE) startet, welches zunächst den Watchdog (Wd) startet, welcher beim Überschreiten der Zykluszeit das Exe-Engine-Objekt (ExE) zurücksetzt, und  
25       anschließend zyklisch
  - die Eingänge des Prozeßabbildes aktualisiert,
  - für den Fall einer zyklischen Bearbeitung des Steuerprogramms jeweils einen Bearbeitungsschritt der Soft-

19.12.95

17

ware-Funktionsbaustein-Objekte (SF01 ... SF04) bearbeitet,

- 5       - für den Fall einer interruptgesteuerten Bearbeitung des Steuerprogramms Änderungen von Signalzuständen an den Eingängen feststellt und die diesen Eingängen zugeordneten Software-Funktionsbaustein-Objekte (SF01 ... SF04) bearbeitet,
- die Ausgänge des Prozeßabbildes aktualisiert.

10   11. Intelligentes Feldgerät nach Anspruch 10, dadurch gekennzeichnet, daß das Exe-Engine-Objekt (ExE) sowie der Watchdog (Wd) als „threads“ ausgebildet sind.

15   12. Intelligentes Feldgerät nach einem der Ansprüche 9 bis 11, dadurch gekennzeichnet, daß die Kommunikationsschnittstelle eine TCP/IP-Protokoll-Kommunikation ermöglicht.

20   13. Intelligentes Feldgerät nach einem der Ansprüche 9 bis 12, dadurch gekennzeichnet, daß die Software-Funktionsbausteine (SF01 ... SF04) Java-bytecodiert sind und in der Programmiersprache „JAVA C“ oder in einer Programmiersprache nach der Norm IEC 1131 erstellbar sind.

14. Automatisierungssystem

- 25   - mit mindestens einem Automatisierungsgerät nach einem der Ansprüche 1 bis 4,
- mit mindestens einem Programmiergerät nach einem der Ansprüche 5 bis 8.

19.12.95

18

15. Automatisierungssystem nach Anspruch 14 mit mindestens einem intelligenten Feldgerät nach einem der Ansprüche 9 bis 13.

5 16. Automatisierungsverbund

- mit einem Automatisierungssystem nach Anspruch 14 oder 15 und

- mit mindestens einer Workstation und/oder einem Server, welche Mittel zum Erstellen und Bearbeiten von objekt-

10 orientierten Software-Funktionsbausteinen (SF01 ... SF04) aufweisen.

19.12.98

1/3

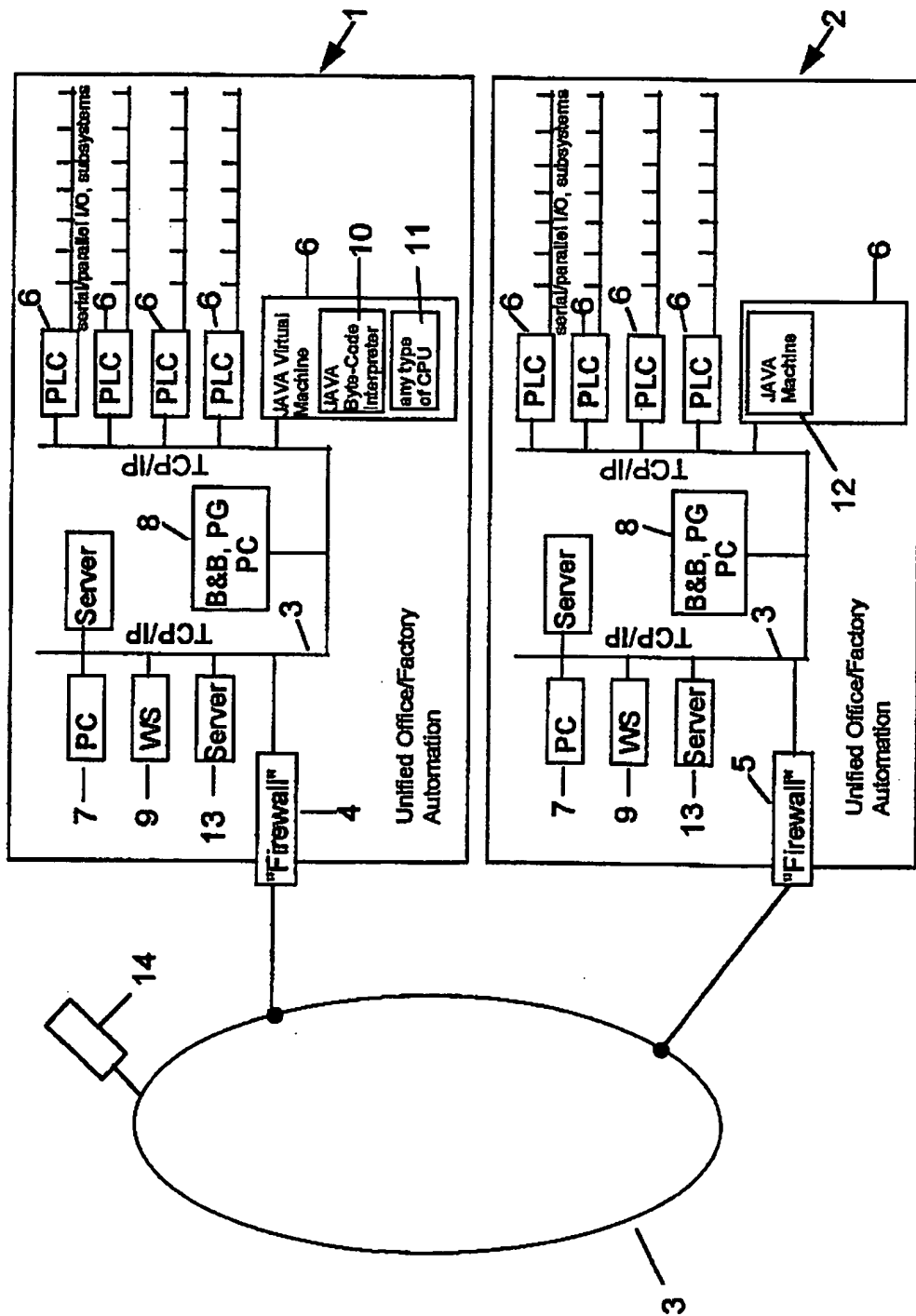


FIG 1

19.12.98

2/3

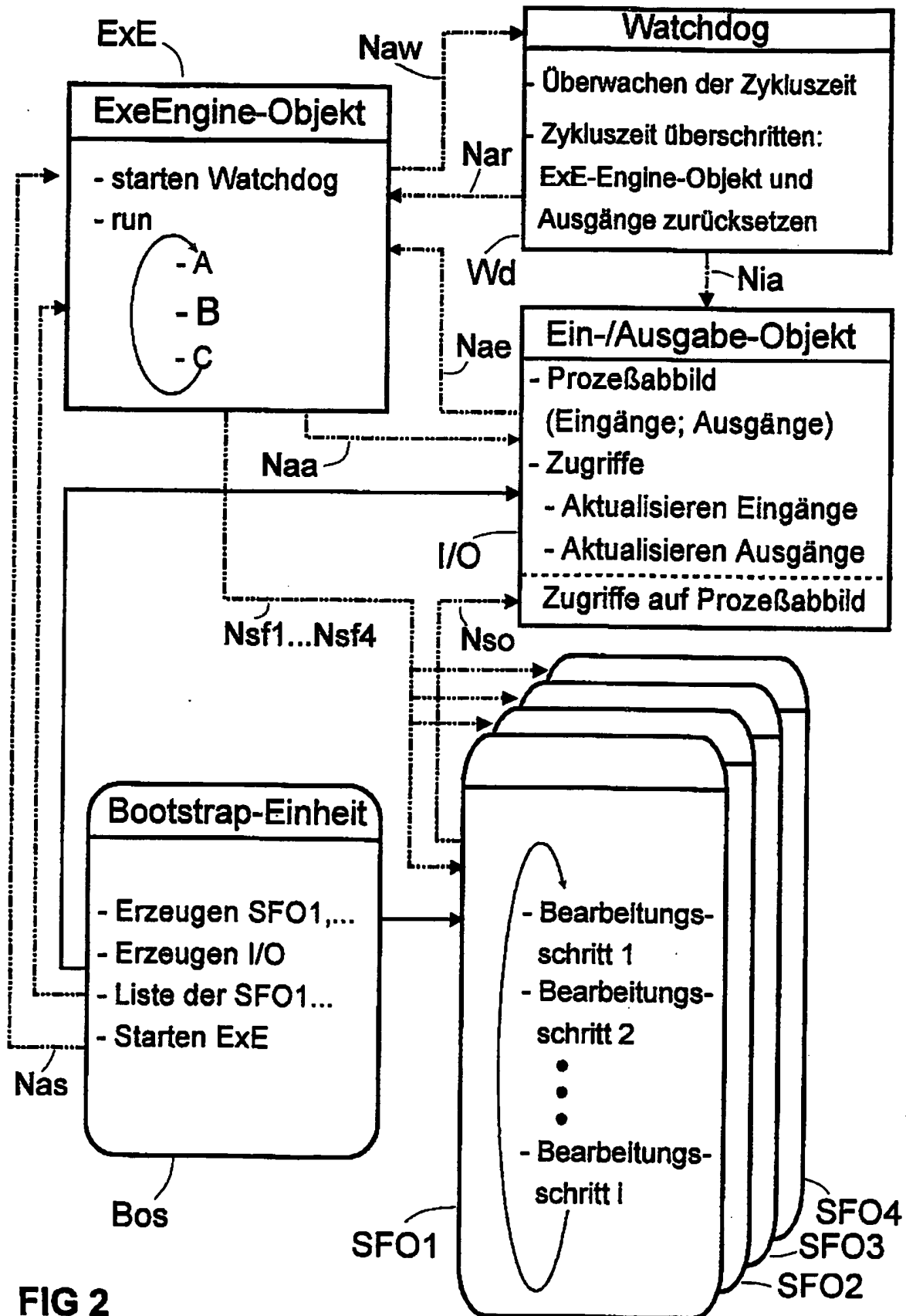


FIG 2

19.12.98

3/3

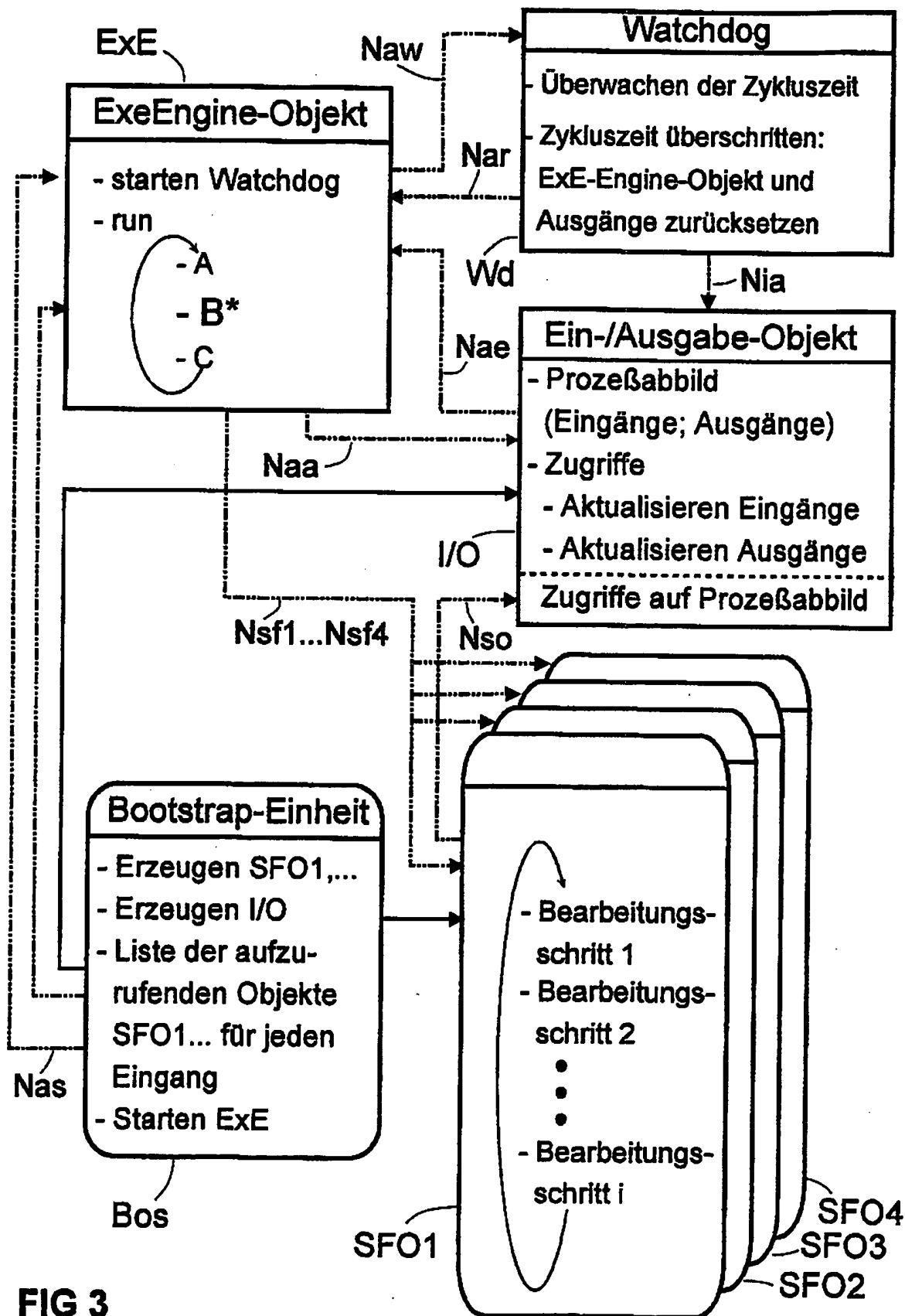


FIG 3